Secure Software Leasing and Implications for Quantum Copy-Protection and Obfuscation

Prabhanjan Ananth
UC Santa Barbara

Joint with
Gorjan Alagic (QuICS, University of Maryland),
Zvika Brakerski (Weizmann Institute of Science),
Yfke Dulek (CWI / QuSoft),
Rolando L. La Placa (MIT) and,
Christian Schaffner (University of Amsterdam / QuSoft)

Merge of

"Impossibility of Quantum Virtual Black-Box Obfuscation of Classical Circuits"
Alagic, Brakerski, Dulek, Schaffner
Link: https://arxiv.org/pdf/2005.06432

and

"Secure Software Leasing"
Ananth, La Placa
Link: https://arxiv.org/abs/2005.05289

Unclonable Cryptographic Primitives

Use no-cloning property of quantum computing
to obtain exciting cryptographic primitives!

Unclonable Cryptographic Primitives

Use no-cloning property of quantum computing
to obtain exciting cryptographic primitives!

Unclonable Cryptographic Primitives

Use no-cloning property of quantum computing
to obtain exciting cryptographic primitives!

- Quantum Money [Weisner'83,...]
- Quantum Copy-Protection [Aar'09]
- Certifiable Deletion Encryption [BI'19]
- Revocable Timed-Release Encryption [Unr'13]
- Unclonable Encryption [Got'02,BL'19]
- One-Shot Signatures and Signature Tokens [BS'16,AGKZ'20]
- Quantum Lightning [Zha'17]

Problem: Software Piracy

Software vendors want to prevent users to generate copies of their software for profit.

Problem: Software Piracy

Software vendors want to prevent users to generate copies of their software for profit.

Classical representation of software can be copied.

Problem: Software Piracy

Software vendors want to prevent users to generate copies of their software for profit.

Classical representation of software can be copied.

Can we use quantum no-cloning theorem to solve this?

Quantum Copy Protection [Aaronson CCC'09]

Quantum Copy Protection [Aaronson CCC'09]

Quantum copy-protection: A solution for preventing software piracy.

Quantum Copy Protection [Aaronson CCC'09]

Quantum copy-protection: A solution for preventing software piracy.

- **Functionality**: The vendor can copy-protect the software and send the copy-protected version to the user.

Quantum Copy Protection [Aaronson CCC'09]

Quantum copy-protection: A solution for preventing software piracy.

- **Functionality**: The vendor can copy-protect the software and send the copy-protected version to the user.
- **Security**: The user shouldn't be able to re-distribute the software to other people.

Quantum Copy Protection [Aaronson CCC'09]

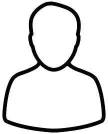
Quantum copy-protection: A solution for preventing software piracy.

- **Functionality**: The vendor can copy-protect the software and send the copy-protected version to the user.
- **Security**: The user shouldn't be able to re-distribute the software to other people.
  (uncloneability!)

Quantum Copy-Protection [Aaronson CCC'09]

Microsoft

(Eval, ρ_C) →

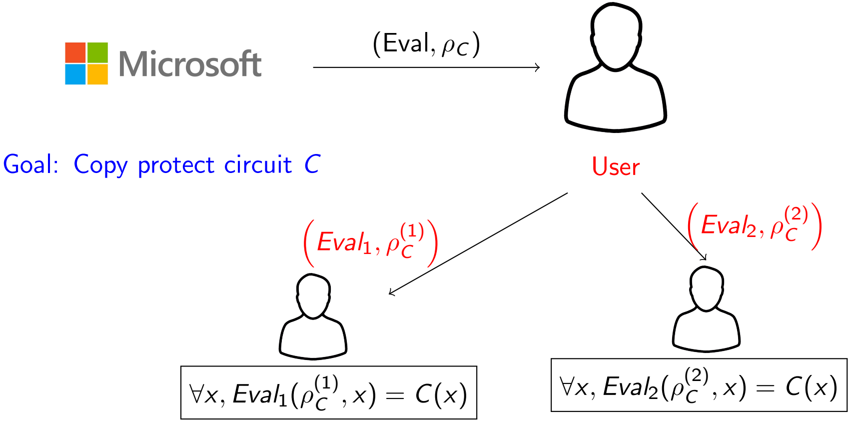


Goal: Copy protect circuit C

User

Functionality: $\forall x, \text{Eval}(\rho_C, x) = C(x)$

Quantum Copy-Protection [Aaronson CCC'09]



User can succeed only with very small (negligible) probability.

(For simplicity: consider product states and only two copies.
Adversary can output entangled states and multiple copies.)

Implications of Quantum Copy-Protection

Implications of Quantum Copy-Protection

- Public-Key Quantum money [ALLZZ'20]
- Unclonable encryption [BL'19]
- Unclonable Decryption Keys [GZ'20]

Quantum Copy-Protection

Quantum Copy-Protection

- Unlearnable functions in classical oracle model
  [Aaronson-Liu-Liu-Zhang-Zhandry arXiv'20]

Quantum Copy-Protection

- Unlearnable functions in classical oracle model
[Aaronson-Liu-Liu-Zhang-Zhandry arXiv'20]

- Heuristic construction for point functions in the plain model
[Aaronson CCC'09]

Does there exist quantum copy-protection for all unlearnable functions?

(open since [Aaronson CCC'09])

Does there exist quantum copy-protection for all unlearnable functions?
(open since [Aaronson CCC'09])

Our work: NO (conditionally)

Our Result

Theorem.

Based on quantum hardness of learning with errors (QLWE), there does not exist quantum copy-protection for all unlearnable functions.

Our Result

Theorem.

Based on quantum hardness of learning with errors (QLWE), there does not exist quantum copy-protection for all unlearnable functions.

Learning with errors is heavily used in cryptography.

It is conjectured to be secure against QPT algorithms

We identify a class of functions such that:

- *This class of functions is quantum unlearnable.*

We identify a class of functions such that:

- *This class of functions is quantum unlearnable.*
- *Given any copy-protected state that computes this function, we can create new copies of this function.*

Implications to Program Obfuscation

Program Obfuscation

A compiler: $C \to \widehat{C}$

- $\widehat{C} \equiv C$ and,
- $\widehat{C}$ hides the implementation details of C.

Obfuscation has powerful implications in crypto and beyond.

- Secure multiparty computation
- Functional encryption
- Delegation
- Instantiating oracles.
- Differential privacy lower bounds
- Hardness of finding Nash
- $\cdots$

Virtual Black-box Property

For any QPT adversary $\mathcal{A}$, there exists a QPT simulator Sim, with oracle access to C, such that:

$$\{\mathcal{A}(\widehat{C})\} \approx \{\mathsf{Sim}^{\mathcal{O}(C)}(1^{|C|})\}$$

Quantum Virtual black-box (qVBB) Obfuscation:
transforms (classical) circuit C into a quantum state ρ_C:

Quantum Virtual black-box (qVBB) Obfuscation:
transforms (classical) circuit C into a quantum state ρ_C:

(i) ρ_C computes C and,

Quantum Virtual black-box (qVBB) Obfuscation:
transforms (classical) circuit C into a quantum state ρ_C:

(i) ρ_C computes C and,

(ii) having ρ_C is the same as having black-box access to C.

Quantum Virtual black-box (qVBB) Obfuscation:
transforms (classical) circuit C into a quantum state ρ_C:

(i) ρ_C computes C and,

(ii) having ρ_C is the same as having black-box access to C.

Our Result

Theorem.

Based on quantum hardness of learning with errors (QLWE), there does not exist qVBB for classical circuits.

Our Result

Theorem.

Based on quantum hardness of learning with errors (QLWE), there does not exist qVBB for classical circuits.

Prior work: ruled out qVBB only for quantum circuits into reusable obfuscated states [AF'16]

Constructing quantum copy-protection for a subclass of unlearnable functions?

(without oracles)

Constructing quantum copy-protection
for
a subclass of unlearnable functions?

(without oracles)

...seems hard.

Constructing quantum copy-protection
for
a subclass of unlearnable functions?
(without oracles)

...seems hard.

However, in some settings, weaker notions suffice.

Copy-protection: given (Eval, ρ_C), adversary cannot produce $(Eval_1, \rho_C^{(1)})$ and $(Eval_2, \rho_C^{(2)})$ such that:

$$Eval_1(\rho_C^{(1)}, \cdot) \text{ computes } C \text{ and,}$$
$$Eval_2(\rho_C^{(2)}, \cdot) \text{ computes } C.$$

In some scenarios, adversary does not get to choose its own evaluation algorithms.

In some scenarios, adversary does not get to choose its own evaluation algorithms.

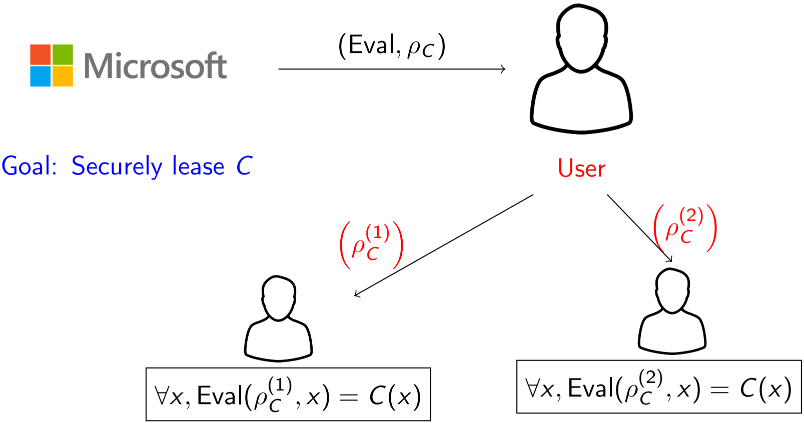
Example: Software X runs only on a specific OS.

Adversary can create "open source" version of X.

But we want to prevent them from creating new copies of X that run on the same OS.

Our Notion: Secure Software Leasing

Secure Software Leasing



User can succeed only with very small (negligible) probability.

(For simplicity: product states. Adversary can output entangled states.)

Adversary receives (ρ_C, Eval).

Infinite-term Lessor Security: only with negl. probability, it can produce $(\rho_C^{(1)})$ and $(\rho_C^{(2)})$ such that,

$$\text{Eval}(\rho_C^{(1)}, \cdot) \equiv C \text{ and,}$$
$$\text{Eval}(\rho_C^{(2)}, \cdot) \equiv C$$

Adversary receives (ρ_C, Eval).

Infinite-term Lessor Security: only with negl. probability, it can produce $(\rho_C^{(1)})$ and $(\rho_C^{(2)})$ such that,

$$\mathsf{Eval}(\rho_C^{(1)}, \cdot) \equiv C \text{ and,}$$
$$\mathsf{Eval}(\rho_C^{(2)}, \cdot) \equiv C$$

Finite-term Lessor Security: adversary has to return back ρ_C. After returning back the state, only with negligible probability, it can produce ρ'_C such that,

$$\mathsf{Eval}(\rho'_C, \cdot) \equiv C$$

Result

Theorem.

*Based on cryptographic assumptions,
there exists infinite-term SSL for a subclass of unlearnable functions.*

Class of functions: compute-and-compare.
($f_a(x)$: take as input x, computes on x to obtain a' and outputs 1 if and only if $a' = a$.)

Subsequent Work

- Finite-term SSL for a subclass of unlearnable functions from QLWE [KNY'20]
- Infinite-term SSL for a different class of unlearnable functions [ALLZZ'20]
- Information-theoeretic SSL in the random oracle model [CMP'20]

Summary of Results

1. Conditional impossibility result of quantum copy-protection
2. Conditional impossibility result of quantum VBB obfuscation.
3. Construction of a weaker notion of copy-protection, called SSL.

Impossibility of Quantum Copy-Protection

Tool: Quantum Fully Homomorphic Encryption (QFHE)

Given encryption of ρ, quantum circuit C,

can efficiently recover encryption of $C(\rho)$.

First Attempt:
Barak et al.'s technique [BGIRSVY CRYPTO'01]

Using QFHE, we construct a class of unlearnable circuits that cannot be copy-protected

Class of Circuits

Class of Circuits

$C_{a,b}$: on input x,

Class of Circuits

$C_{a,b}$: on input x,

- If $x = 0$, output $\mathrm{Enc}(a)$.

Class of Circuits

$C_{a,b}$: on input x,

- If $x = 0$, output $\mathrm{Enc}(a)$.
- If $x = a$, output b.

Class of Circuits

$C_{a,b}$: on input x,

- If $x = 0$, output $\text{Enc}(a)$.
- If $x = a$, output b.
- On all other inputs, output 0.

Class of Circuits

$C_{a,b}$: on input x,

- If $x = 0$, output $\mathsf{Enc}(a)$.
- If $x = a$, output b.
- On all other inputs, output 0.

Proof of quantum unlearnability:
Adversary method [Ambainis STOC'00]

Class of Circuits

$C_{a,b}$: on input x,

Class of Circuits

$C_{a,b}$: on input x,

- If $x = 0$, output $\mathsf{Enc}(a)$.

Class of Circuits

$C_{a,b}$: on input x,

- If $x = 0$, output $\mathsf{Enc}(a)$.
- If $x = a$, output b.

Class of Circuits

$C_{a,b}$: on input x,

- If $x = 0$, output $\mathsf{Enc}(a)$.
- If $x = a$, output b.
- On all other inputs, output 0.

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

- On input 0, obtain $\mathrm{Enc}(a)$.

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

- On input 0, obtain $\mathrm{Enc}(a)$.
- Homomorphically evaluate U on input $\mathrm{Enc}(a)$ and $\mathrm{Enc}(\rho_C)$.
  The result is $\mathrm{Enc}(U(\rho_C, a)) = \mathrm{Enc}(|b\rangle\langle b| \otimes \rho'_C)$

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

- On input 0, obtain $\mathsf{Enc}(a)$.
- Homomorphically evaluate U on input $\mathsf{Enc}(a)$ and $\mathsf{Enc}(\rho_C)$.
  The result is $\mathsf{Enc}(U(\rho_C, a)) = \mathsf{Enc}(|b\rangle\langle b| \otimes \rho'_C)$

To copy $C_{a,b}$, we need to recover b...

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

- On input 0, obtain $\text{Enc}(a)$.
- Homomorphically evaluate U on input $\text{Enc}(a)$ and $\text{Enc}(\rho_C)$.
  The result is $\text{Enc}(U(\rho_C, a)) = \text{Enc}(|b\rangle\langle b| \otimes \rho'_C)$

To copy $C_{a,b}$, we need to recover b...
.. but b is encrypted.

Idea: use special-purpose program obfuscation to recover b.

This notion can be based on QLWE.

Class of Circuits

$C_{a,b}$: on input x,

- If $x = 0$, output $(\mathrm{Enc}(a),\ \mathcal{O}(G))$
- If $x = a$, output b.
- On all other inputs, output 0.

Description of G:

On input $\mathrm{Enc}(b)$, output a, b
(implicity the function G has the decryption key hardwired inside it.)

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

- On input 0, obtain Enc(a).

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

- On input 0, obtain $\mathsf{Enc}(a)$.
- Homomorphically evaluate U on input $\mathsf{Enc}(a)$ and $\mathsf{Enc}(\rho_C)$.
  The result is $\mathsf{Enc}(U(\rho_C, a)) = \mathsf{Enc}(|b\rangle\langle b| \otimes \rho'_C)$

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

- On input 0, obtain $\mathsf{Enc}(a)$.
- Homomorphically evaluate U on input $\mathsf{Enc}(a)$ and $\mathsf{Enc}(\rho_C)$.
  The result is $\mathsf{Enc}(U(\rho_C, a)) = \mathsf{Enc}(|b\rangle\langle b| \otimes \rho'_C)$
- Compute $\mathcal{O}(G)$ on $\mathsf{Enc}(a), \mathsf{Enc}(|b\rangle\langle b| \otimes \rho'_C)$ to recover a, b.

Insecurity of copy-protection of $C_{a,b}$

Given (U, ρ_C) implementing $C_{a,b}$, do the following:

- On input 0, obtain $\mathrm{Enc}(a)$.
- Homomorphically evaluate U on input $\mathrm{Enc}(a)$ and $\mathrm{Enc}(\rho_C)$.
  The result is $\mathrm{Enc}(U(\rho_C, a)) = \mathrm{Enc}(|b\rangle\langle b| \otimes \rho'_C)$
- Compute $\mathcal{O}(G)$ on $\mathrm{Enc}(a), \mathrm{Enc}(|b\rangle\langle b| \otimes \rho'_C)$ to recover a, b.

Using $\mathrm{Enc}(a), a, b$, can create as many copies of $C_{a,b}$ as we want!

Requirement of QFHE: evaluation of arbitrary depth quantum circuits

Requirement of QFHE: evaluation of arbitrary depth quantum circuits

Current constructions of QFHE based on circular-secure QLWE [Mahadev FOCS'18, Brakerski CRYPTO'18]

Removing Circular Security

Replace QFHE with leveled QFHE.
(Leveled QFHE: evaluation of fixed-depth quantum circuits.)

Idea

Instead of producing the public-key and the ciphertext in one shot,

produce the public-key and the ciphertext gradually over many computations of the circuit.

(Each computation produces a small piece of the public-key and the ciphertext)

Removing Circular Security

Replace QFHE with leveled QFHE.

(Leveled QFHE: evaluation of fixed-depth quantum circuits.)

Idea

Instead of producing the public-key and the ciphertext in one shot,

produce the public-key and the ciphertext gradually over many computations of the circuit.

(Each computation produces a small piece of the public-key and the ciphertext)

Refer to [Alagic-Brakerski-Dulek-Schaffner'20] for more details.

Summary and Future Questions

Summary:

- For contrived class of unlearnable functions, copy-protection is impossible.
- Weaker notions of copy-protection for a non-trivial class of unlearnable functions can be constructed.
  (first feasibility results in the plain model)

Summary and Future Questions

Summary:

- For contrived class of unlearnable functions, copy-protection is impossible.
- Weaker notions of copy-protection for a non-trivial class of unlearnable functions can be constructed.
  (first feasibility results in the plain model)

Future Questions:

- Other variants of copy-protection.
- Provably secure constructions copy-protection for any non-trivial class of circuits (such as point functions).
- Constructing weaker variants of quantum obfuscation.
  For example: quantum indistinguishability obfuscation.

Thanks! [1]

[1] Some of the slides were prepared by Rolando L. La Placa